Методичка 5.4 - Перенаправление сетевых запросов

Перенаправление сетевых запросов

В предыдущих видео мы работали с программой prog-4.1, которая самостоятельно выполняет проверку ключа, но существуют программы, которые выполняют проверку ключа на удаленном сервере, который находится где-то в сети Интернет.

Обращение к таким серверам обычно осуществляется по доменному имени. Протокол для передачи данных может быть любой. От обычного HTTP до проприетарного, который был придуман разработчиками программы.

Для демонстрации того, как можно выполнить активацию программы путем перенаправления сетевых запросов, была подготовлена программа prog-4.4.1, которая выполняет проверку ключа на удаленном сервере. Это ее единственное отличие от программы prog-4.1.

Давайте попробуем запустить программу prog-4.4.1, убедиться в том, что она работает правильно.

> prog-4.4.1.exe 1 2

> prog-4.4.1.exe 1 3

Мы видим, что счетчик счет начал уменьшаться. Давайте попробуем выполнить активацию программы.

> prog-4.4.1.exe blabla

Request has been sent to license server.

Waiting response from license server…done!

License key is incorrect.

Как видно, ключ не прошел проверку. Давайте разбираться с протоколом взаимодействия программы и удаленного сервера.

Для того, чтобы получить получить доступ к сетевым пакетам нам понадобится сниффер. Я рекомендую использовать Wireshark.

Рекомендации по установке сниффера Wireshark.

Ссылка: https://1.eu.dl.wireshark.org/win64/Wireshark-win64-3.0.3.exe

Запускаем сниффер. Выбираем сетевой интерфейс, через который мы выходим в сеть Интернет.

Видим, что через сетевой интерфейс проходит очень много сетевых пакетов. Сейчас хорошо бы добавить фильтр, но мы пока ничего не знает о протоколе взаимодействия. TCP это или UDP. Используется ли доменное имя или обращение происходит сразу по IP-адресу.

Если программа взаимодействует с удаленным сервером в сети Интернет, то у нее точно в коде или каком-нибудь конфиге должен быть указан IP-адрес или доменное имя удаленного сервера, иначе она не будет знать, как с ним связаться.

В нашем случае программа prog-4.4.1.exe состоит из одного файла. Давайте с помощью утилиты Strings получим все строки, которые есть в бинарном файле и попробуем найти что-нибудь похожее на IP-адрес или доменное имя.

Рекомендации по установке утилиты Strings.

Ссылка: https://docs.microsoft.com/en-us/sysinternals/downloads/strings

Запускаем утилиту Strings.

> strings64.exe prog-4.4.1.exe > out.txt

Открываем файл out.txt и видим более 2 тыс. строк. Это очень много для того, чтобы искать в ручном режиме. Давайте сократим область поиска.

Если доменное имя есть в составе программы, то оно находится в секции данных. Запускаем утилиту Puppy. Открываем файл prog-4.4.1.exe. Выбираем Section Headers. Видим, что секция данных в файле начинается со смещения 0x17a00.

Утилита Strings принимает смещение только в десятичной системе счисления, поэтому открываем калькулятор и переводим число из шестнадцатиричной в десятичную.

Получаем число 96768.

Запускаем снова утилиту Strings.

> strings64.exe -f 96768 prog-4.4.1.exe > out.txt

Открываем файл out.txt и видим всего 42 строки. И почти сразу находим доменное имя check-key.example.com. Это говорит о том, что программа, скорее всего, обращается к серверу по доменному имени. Значит перед тем, как отправить запрос на проверку ключа, происходит разрешение доменного имени для получения IP-адреса. А для разрешения доменных имен используется протокол DNS.

Возвращаемся к снифферу и добавляем фильтр по протоколу DNS.

dns

Давайте еще раз попробуем активировать нашу программу и убедимся в том, что действительно доменное имя check-key.example.com используется программой при активации.

> prog-4.4.1.exe blabla

В сетевом трафике мы быстро находим DNS-запрос. В ответе мы видим, что этому доменному имени соответствует IP-адрес 192.168.1.8.

Давайте посмотрим, какие сетевые пакеты были отправлены по этому IP-адресу.

dns || ip.dst == 192.168.1.5

Мы видим, что было установлено TCP-соединение с сервером и в одном из пакетов отправлен ключ в открытом виде - “blabla”. Обратите внимание, что порт назначения 1337, это нам понадобится позже.

В ответ была отправлена строка “false” и соединение завершилось. Мы видим, что никаких механизмов контроля целостности сообщений не используется. Мы знаем, что в случае неправильного ключа сервер отвечает строкой “false”, но при этом не знаем, что он отвечает, когда ключ правильный.

Давайте откроем программу prog-4.4.1.exe под отладчиком и найдем код, который обрабатывает ответы от сервера.

Переходим в секцию кода и пробуем найти код, который ссылается на строку “License key is incorrect”. Поставим точку останова перед условным переходом и перезапустим программу в режиме активации.

Функция 0x00401050 вернула нулевое значение через регистр EAX. Давайте перейдем внутрь этой функции, поставим точку останова и перезапустим программу.

Видим, что внутри функции происходит отправка сетевых пакетов, далее мы получаем ответ от сервера и выполняем сравнение двух строк с помощью функции strcmp.

Сравнение ответа происходит со строкой “true”. А если мы указываем при активации программы ключ “blabla”, то мы получаем от сервера строку “false”.

Наша задача сделать так, чтобы сервер прислал строку “true”.

Давайте выполним перенаправление запроса на наш сервер.

Специального для этой цели была разработана программа prog-4-4.2, которая умеет отвечать на все запросы строкой “true”.

Запускаем наш сервер:

> prog-4.4.2.exe

Итак, сейчас сервер работает локально на нашем компьютере, теперь нам надо перенаправить запрос на наш сервер.

Это можно очень легко сделать, если добавить следующую запись в файл C:\Windows\System32\hosts

127.0.0.1 check-key.example.com

Проверим, что теперь доменное имя check-key.example.com разрешается IP-адресом 127.0.0.1.

Открываем командную строку

> ping check-key.example.com

Видим, что теперь доменному имени соответствует IP-адрес 127.0.0.1.

Давайте попробуем активировать программу.

> prog-4.4.1.exe blabla

The program has been activated

Откроем консоль, где был запущен сервер. Видим, что он принял запрос и ответил строкой “true”.

Открываем сниффер и видим, что новых запросов на настоящие сервер не выполнялось.